**Lab18\_Strings\_1**

**Problem Description:** The student name and the branch name details of a particular student

must be captured and validated.

Add the following mentioned properties and methods from the below class diagram to the existing Student class ***from lab4***



**Step 1: copy and** Modify the ***Student*** class created (Student.java) in Lab4 to include the

instance variables and the methods whose prototype is given above:

**Step 2:** The logic for the ***validateStudentName()***is as follows:

a. Check if the length of the ***studentName*** is between 5 and 25(both inclusive). If

not, display a message “The length of the name should be minimum 5 characters

and a maximum of 25 characters” and return false. Otherwise, return true.

**Step 3:** The logic for the ***validateBranchName()***is as follows:

a. Using the following table, check whether the ***branchName*** is valid. Do a case

insensitive comparison of Strings.

**Table 3**

**branchName**

CSE

ECE

EEE

MECH

Bio-Tech

If it is valid, return true, else set the value of ***branchName*** to CSE and display a

message “Invalid Branch Name, set to CSE” and return false.

**Step 4:** Add the following statements to the existing main method written in ***Demo*** class

(Demo.java in Lab 11)

Create a reference variable of Student class(or use the existing object created earlier) with

the name **studentOne** and instantiate the same

a. Invoke the corresponding setter methods to set the values for the instance

variables as follows:

**Table 4**

studentId 1001

studentName Jackson

qualifyingExamMarks 95.0f

residentialStatus D

yearOfEngg 2

branchName CSE

b. Invoke the ***validateStudentName()*** and ***validateBranchName()*** methods. If the

return types of both the methods are true,

i. Invoke the ***validateExamMarks()*** method

ii. If the ***qualifyingExamMarks*** is invalid, display “Invalid marks, the range of

marks is between 65 and 100”

iii. If the ***qualifyingExamMarks*** is valid,

Using the corresponding getter method, display the details as follows:

Student Id :\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Student Name :\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Qualifying Marks :\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Residential Status :\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Current Year of Engineering :\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Branch Name :\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Step 5:** Compile the program, fix the errors if any

**Step 6:** Execute the program and verify the output

**Step 7:** Create reference variables of Student class with the name **studentTwo** and

instantiate the same

a. Invoke the corresponding setter methods to set the values for the instance

variables as follows:

**Table 5**

studentId 1002

studentName Jen

qualifyingExamMarks 68.0f

residentialStatus H

yearOfEngg 3

branchName ABC

Repeat **Step 4.b** for **studentTwo** created in **Step 7**, compile and execute the program. Do you

find a difference in the output? Why?

**At the end of the assignment you would have the following files:**

Student.java

Demo.java